The opinion in support of the decision being entered today was not written for publication and is not binding precedent of the Board.

Paper No. 30

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES

Ex parte TING YU LEUNG,
DAVID E. SIMMEN, and YANG SUN

MAILED

MAY 20 2005

U.S. PATENT AND TRADEMARK OFFICE
BOARD OF PATENT APPEALS
AND INTERFERENCES

Appeal No. 2004-1538
Application 09/502,820[1]

ON BRIEF

Before THOMAS, KRASS, and BARRETT, Administrative Patent Judges.

BARRETT, Administrative Patent Judge.

DECISION ON APPEAL

This is a decision on appeal under 35 U.S.C. § 134(a) from the final rejection of claims 1-5, 12-16, and 23-27. Claims 6, 17, and 28 have been canceled. Claims 7-11, 18-22, and 29-60 have been objected to as depending from a rejected base claim.

We reverse.

[1] Application for patent filed February 11, 2000, entitled "Optimizing Database Queries Using Query Execution Plans Derived From Automatic Summary Table Determining Cost Based Queries," which is based on and claims priority under 35 U.S.C. § 119(e)(1) from U.S. Provisional Application 60/134,745, filed May 18, 1999.

BACKGROUND

The invention relates to the optimization of queries using automatic summary tables in a database management system.

Claim 1 is reproduced below.

> 1. A method of optimizing a query in a computer, the query being performed by the computer to retrieve data from a database stored on the computer, the method comprising the steps of:
>
> (a) identifying one or more automatic summary tables (ASTs) that overlap the query by matching definitions of the ASTs with requirements of the query, wherein the requirements of the query satisfied by an AST are encapsulated as a set of properties;
>
> (b) enumerating one or more alternative query execution plans (QEPs) for the query, including at least one QEP that represents one or more access paths of the identified ASTs;
>
> (c) assigning a cost to each of the alternative QEPs; and
>
> (d) choosing a most efficient one of the alternative QEPs based upon the assigned costs.

THE REFERENCES

The examiner relies on the following references:

Osborn et al. (Osborn)	6,026,391	February 15, 2000 (filed October 31, 1997)
Agarwal et al. (Agarwal)	6,370,522	April 9, 2002 (filed March 18, 1999)

THE REJECTIONS

Claims 1-3, 12-14, and 23-25 stand rejected under 35 U.S.C. § 102(a, e) by Osborn.

Claims 4, 5, 15, 16, 26, and 27 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Osborn and Agarwal.

We refer to the final rejection (Paper No. 19) (pages referred to as "FR__") and the examiner's answer (Paper No. 26) (pages referred to as "(EA__") for a statement of the examiner's rejection, and to the brief (Paper No. 25) (pages referred to as "Br__") and reply brief (Paper No. 27) (pages referred to as "RBr__") for a statement of appellants' arguments thereagainst.

OPINION

The invention

It will be helpful to discuss the disclosed invention, in particular, some of the terminology that appears in the claims. An "automatic summary table (AST)" is a table resulting from executing a query where the definition of the summary table is based on a "full select" statement (page 2, lines 9-13; page 29, lines 19-21). An AST example is the rich-employee AST which is defined by a SQL statement for records of employees which make over $100,000 (page 10; Table 1, page 15). In the prior art, the AST could not be used by an optimizer to improve performance (page 2, lines 6-8). A "query" means to interrogate a collection of data such as records in a database. A query has a set of "requirements" that usually include at least the table or tables is the data coming from; the selection criteria, which is the matching condition or filter; and, which columns or fields in the tables are to be displayed or printed in the result. An example of query requirements is shown in Table 3 (page 20).

The limitation of "identifying one or more automatic summary tables (ASTs) that overlap the query by matching definitions of the ASTs with requirements of the query" can be explained by example. Table 1 (page 15) shows the properties and matching information associated with the rich-employees AST, with one predicate of "SALARY > 100,000," and Query 1 has predicates "SALARY > 100,000" and "LOCATION = PITTSBURGH," so the rich-employees AST overlaps Query 1 as to "SALARY > 100,000" (page 24, lines 16-23). That is, the rich-employees AST was defined using "SALARY > 100,000" and the matching step determines whether the AST was derived in such a way that it can be used as a starting point to satisfy the query (page 2, lines 18-20).

The limitation that "the requirements of the query satisfied by an AST are encapsulated as a set of properties" is shown by the properties in Table 1 (page 15), where the AST "properties" resulting from the matching phase include relational properties such as the tables referenced, columns supplied, expressions computed, predicates applied, unique keys and functional dependencies in effect, aggregation performed, and so on (page 14, lines 6-8).

A "query execution plan (QEP)" specifies the sequence of database operations used to satisfy the query (page 17, line 7). Each QEP represents an "access path" to a table or an AST. For example, QEP1 could represent an "index access" to the employee

table, where the index is on the salary column and is used with the predicate "SALARY > 100,000" to directly access records of employees whose salaries exceed 100,000 (Fig. 4A; page 18, line 21, to page 19, line 6) and QEP2 could represent a "full table scan" of the employee table where each record is accessed and qualified against the predicates "SALARY > 100,000" and "LOCATION = PITTSBURGH." The QEPs can be assigned a cost and the most efficient QEP can be selected.

<u>The rejection and arguments</u>

Appellants argue that Osborn does not teach "identifying one or more automatic summary tables (ASTs) that overlap the query by matching definitions of the ASTs with requirements of the query, wherein the requirements of the query satisfied by an AST are encapsulated as a set of properties." The examiner finds that the automatic summary table (AST) corresponds to Osborn's summary table detailed in Fig. 3, element 68, and column 6, lines 51-64, and finds that the matching limitation is taught at column 6, lines 20-35 (FR6; EA5). The examiner states that summary tables are an integral part of Osborn's database and that summary tales are shown in Fig. 3 and that Osborn teaches transmitting an SQL statement to a cost optimizer module which considers access paths to the requested data and estimates the relative cost of each execution plan (EA10-11). Appellants argue that Osborn merely describes generating a set of potential execution plans based on

the available access paths, and estimating the cost of each potential execution plan based on the data distribution and storage characteristics for the respective tables, clusters, and indexes to be used (Br6; RBr4). It is argued that nothing in Osborn describes matching definitions of the ASTs with requirements of the query in order to identify the ASTs that overlap the query and encapsulating the requirements of the query satisfied by an AST as a set of properties (Br6; RBR4).

Appellants also argue that Osborn does not teach "enumerating one or more alternative query execution plans (QEPs) for the query, including at least one QEP that represents one or more access paths of the identified ASTs." The examiner relies on column 6, lines 14-30, where it is said that Osborn teaches transmitting a SQL statement to a cost optimizer module that is associated with the database element 34 in Fig. 2, where the cost optimizer element 42 considers access paths to the requested data from the database and derives a plan for executing the query that is the most efficient as described at column 6, lines 23-29 (FR6; EA5). The examiner states that Osborn is directed to estimating query response time for database inquiries based on access paths (EA11). Appellants argue that Osborn merely describes generating a set of potential execution plans based on the available access paths of the query, but not based on access paths of ASTs identified as overlapping the query (Br6; RBr4).

Analysis

We agree with appellants that Osborn does not expressly or impliedly teach limitations (a) and (b) of claim 1. The relevant part of Osborn states (col. 6, lines 16-41):

> The cost optimizer 42 considers available access paths to the requested data from the database 32, and derives a plan for executing the query that is most efficient based on statistics maintained in a data dictionary associated with the respective table(s), along with their associated clusters and indexes, accessed by the SQL statement representing the query 40.
>
> In particular, the cost optimizer 42 generates a set of potential execution plans (not shown) for executing the respective query SQL statement (40) based on the available access paths, and estimates the relative "cost" of each potential execution plan based on the data distribution and storage characteristics for the respective tables, clusters and indexes to be used....
>
> The cost optimizer 42 compares the estimated costs of the potential execution plans and returns the smallest estimated cost 44, along with a result set 45 representing the selected execution plan for the input query, to a query performance prediction ("QPP") module 46 ....

Osborn does not teach the limitation of "(a) identifying one or more automatic summary tables (ASTs) that overlap the query by matching definitions of the ASTs with requirements of the query, wherein the requirements of the query satisfied by an AST are encapsulated as a set of properties" in claim 1. The table in Fig. 3 of Osborn, relied upon by the examiner, is a query history table that is used in assigning a cost; it is not an AST and is not an AST used in determining a QEP. While there is some determination of a match between the table in Fig. 3 and a query

(col. 7, lines 10-12), this is not an overlap between an AST and a query, nor does it meet the limitation that "the requirements of the query satisfied by an AST are encapsulated as a set of properties." Osborn's statement that "the cost optimizer 42 generates a set of potential execution plans (not shown) for executing the respective query SQL statement (40) based on the available access paths" (col. 6, lines 23-26) teaches "enumerating one or more alternative query execution plans (QEPs) for the query," but does not teach "including at least one QEP that represents one or more access paths of the identified ASTs" because it does not teach identifying ASTs that overlap the query. Osborn does disclose "assigning a cost to each of the alternative QEPs" and "choosing a most efficient one of the alternative QEPs based upon the assigned costs," except that the QEPs do not "includ[e] at least one QEP that represents one or more access paths of the identified ASTs" as recited in limitation (b). For these reasons, we find that the subject matter of independent claims 1, 12, and 23 is not anticipated. The rejection of claims 1-3, 12-14, and 23-25 is reversed.

Agarwal does not cure the deficiencies of Osborn with respect to the independent claims. Accordingly, the obviousness rejection of claims 4, 5, 15, 16, 26, and 27 is reversed.

In summary, rejections of claims 1-5, 12-16, and 23-27 are reversed.

REVERSED

JAMES D. THOMAS)
Administrative Patent Judge)

ERROL A. KRASS) BOARD OF PATENT APPEALS AND INTERFERENCES
Administrative Patent Judge)

LEE E. BARRETT)
Administrative Patent Judge)

GATES & COOPER LLP
HOWARD HUGHES CENTER
6701 CENTER DRIVE WEST, SUITE 1050
LOS ANGELES, CA 90045